



Noções de XML

Henrique Silva

Direção-Geral do Território

FCUL, 7 e 14 de Outubro de 2021



MINISTÉRIO DO AMBIENTE, DO ORDENAMENTO DO TERRITÓRIO E ENERGIA
Direção-Geral do Território

O que é o XML? Características principais

- XML significa Extensible Markup Language
- XML é uma linguagem de **marcadores (tags)** à semelhança do HTML
- XML foi desenhado para **descrever dados**
- As tags de XML não são pré-definidas. O utilizador deve **definir as suas próprias tags**.
- XML utiliza uma **Document Type Definition (DTD)** ou um **XML Schema** para descrever e validar os dados.
- **XML com uma DTD ou XML Schema é desenhado para ser auto-descritivo**
- XML é uma Recomendação W3C desde 10 de Fevereiro de 1998.

XML e HTML

A diferença principal entre XML e HTML consiste no facto de o XML ter sido desenhado para **transportar** dados. O XML não é um substituto do HTML, pois ambos foram desenhados tendo em vista objectivos distintos.

XML foi desenhado para **descrever** os dados e está focalizado naquilo que são os dados.

HTML foi desenhado para exibir os dados e está focalizado no modo como os dados são apresentados.

HTML tem como objectivo exibir a informação e formatá-la, enquanto o XML tem como objectivo descrever a informação. XML foi concebido para estruturar, armazenar e enviar informação.

No entanto, um documento XML não faz nada. É apenas informação contida entre tags XML. É necessário escrever software (ou utilizar software preexistente) para enviar, receber ou exibir a informação.

Exemplo de documento XML

À semelhança do HTML, todos os documentos XML devem ter um elemento **root**, que é o elemento que contém todos os outros elementos e marcadores do documento.

Exemplo de uma nota, armazenada como XML:

```
<note>  
<to>Rita</to>  
<from>Pedro</from>  
<heading>Reminder</heading>  
<body>Entregar o relatório até dia 28</body>  
</note>
```

A nota tem um título e um corpo contendo uma mensagem. Tem também informação acerca do emissor e do receptor. O elemento *root* deste exemplo é constituído pelo par `<note></note>`.

Browsers e parsers

Os ficheiros XML podem ser visualizados em browsers tais como: Chrome, Firefox, Opera ou Internet Explorer. No entanto, para conseguir com que os documentos XML sejam exibidos de forma semelhante às páginas Web usuais é necessário acrescentar informação adicional para formatação. Esta informação pode ser codificada no próprio documento XML que se pretende exibir, ou em ficheiros separados.

Os browsers que suportam XML têm que conter uma aplicação designada por XML parser, ou processador XML. A função mais básica do parser é decompor um documento XML nos seus elementos individuais. No seu nível de implementação mínimo, o parser deve ser capaz de percorrer a estrutura em árvore do documento e confirmar que este respeita as cinco regras que caracterizam o chamado XML bem formado.

XML e XSL

O XML sozinho é insuficiente para ter utilidade, diferentemente do HTML. Codificar um documento XML apresenta um nível de dificuldade semelhante ao de codificar uma página HTML; a diferença torna-se notória quando se usa a informação XML.

Um documento XML tem um aspecto semelhante num editor de texto ou num browser. Tal deve-se ao facto de o XML ser informação pura - não tem capacidades de formatação.

Para um documento XML ter uma aparência semelhante a um documento HTML formatado é necessário fornecer instruções separadamente. Actualmente existem dois modos de formatar XML: com Cascading Stylesheets (CSS) ou com XML Stylesheet Language (XSL)

CSS é correntemente usado por programadores de HTML, enquanto XSL é uma tecnologia pertencente à família XML.

Extensibilidade e Independência

XML é livre e extensível

As tags XML não são pré-definidas. Devem ser “inventadas” pelo utilizador.

As tags usadas em documentos HTML bem como a estrutura destes, são pré-definidas. Por exemplo as tags `<i>`, ``, etc.,... são definidas pela Norma HTML.

Em XML o utilizador pode definir as suas próprias tags bem como a estrutura dos seus documentos. Tags como as utilizadas no exemplo anterior, `<note>`, `<to>`, etc.,... são criadas pelo utilizador, não estando definidas em qualquer norma XML.

Sintetizando, uma descrição possível do XML poderia ser:

XML é uma ferramenta para transmissão de informação, multiplataforma, independente em relação ao software e hardware.

Conceitos básicos do XML

- Marcadores (*tags*)
- Elementos
- XML bem formado
- Declaração de documento XML
- Estrutura em árvore de um documento XML bem formado

Marcadores (*tags*)

Tags são linhas de código especiais escritas entre os símbolos “<” e “>”, que instruem os interpretadores HTML ou XML acerca do modo como o texto deve ser exibido ou processado.

Por exemplo: <note> e </from> são tags.

Cada tag é um componente de abertura ou fecho.

Ex: <note> tag de abertura

</note> e </from> tags de fecho

Elementos

Elementos são pares de tags com conteúdo. Cada elemento é uma unidade de informação definida para processamento.

Embora os elementos sejam constituídos pelas tags e pelo conteúdo, cada elemento é considerado como sendo um objecto. Os elementos podem possuir **atributos** que são especificados no interior da tag de abertura. Elementos podem ser constituídos por uma única tag, dizendo-se neste caso que a tag é vazia.

Exemplos de elementos:

1)<to>Rita</to>

2)<from>Pedro</from>

3)<note><to>Rita</to><from>Pedro</from><heading>Reminder</heading>
<body>Entregar o relatório até dia 28</body></note>

XML bem formado

Diz-se que um documento XML é bem formado se estiver em conformidade com as seguintes **5 condições**:

1. As tags têm que ser explícitas. Para cada tag de abertura tem que existir a correspondente tag de fecho. Reciprocamente, para cada tag de fecho tem que existir a correspondente tag de abertura.
2. As tags vazias têm que incluir uma barra (/) antes do último carácter.
Ex: `
`
3. Todos os valores dos atributos devem sempre ser escritos entre aspas.
Ex: `<TD WIDTH="25%">`

XML bem formado

4. **Aninhamento.** Todos os pares devem ser correctamente fechados. Nenhum par pode fechar imediatamente após a abertura de outro diferente.

`<nome>Filipe<sobrenome>Folque</nome></sobrenome>`

Incorrecto

`<nome >Filipe< sobrenome >Folque</ sobrenome ></nome >` **Correcto**

5. As tags são **case sensitive**

`<serie>SCN 10K</SERIE>`

Incorrecto

`<serie>SCN 10K</Serie>`

Incorrecto

XML bem formado

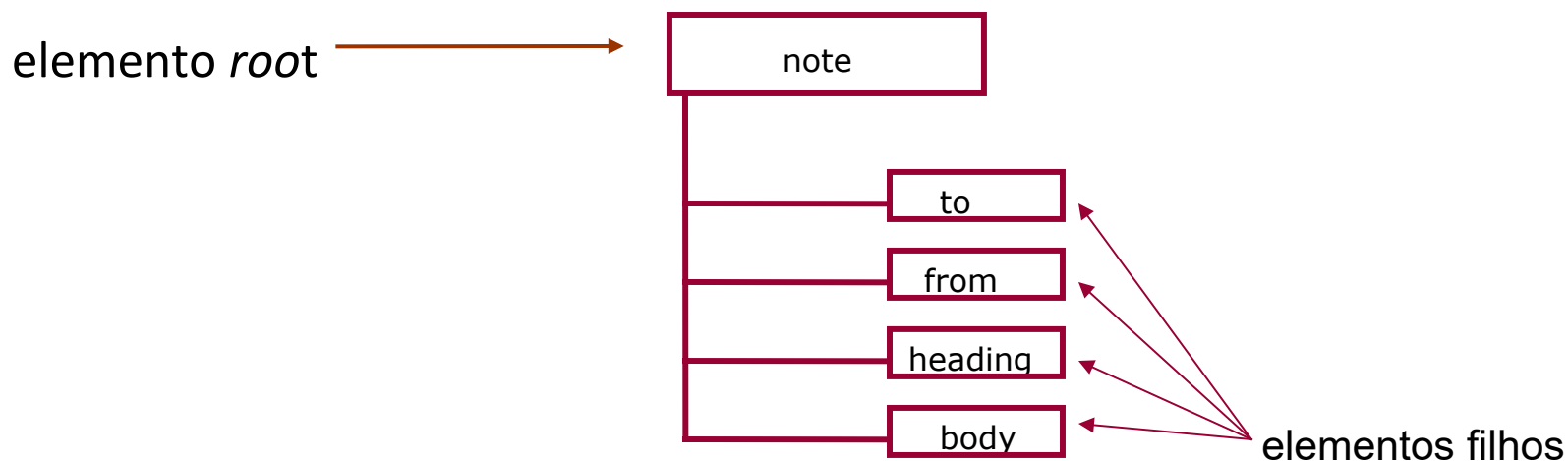
Conclusão:

Um bloco de código XML tem que ser bem formado para poder ser um documento XML. Um bloco que não seja bem formado produzirá uma mensagem de erro no browser, e não pode ser considerado um documento XML.

O XML é muito mais restrictivo que HTML, visto que o não cumprimento de uma das cinco regras enunciadas erro impede uma correcta exibição do código no browser.

Estrutura em árvore de um documento XML bem formado

Um documento XML bem formado cria uma estrutura natural em árvore subordinada ao elemento *root*. Esta representação hierárquica dos elementos facilita uma boa compreensão da estrutura do documento.



Cada um dos elementos filhos num documento XML arbitrário, pode ter os seus próprios elementos filhos, e assim sucessivamente.

Namespaces em XML

Cada programador de XML pode criar o seu próprio conjunto de tags usando quaisquer nomes e atributos que desejar. Existe a possibilidade de o XML poder tornar-se menos normalizado e mais caótico. Neste caso, o resultado estaria em contradição com os objectivos que presidiram à criação do XML.

Ex: Programador A cria a tag: <prego> (descreve uma forma de alimento)

Programador B cria a tag: <prego> (descreve um objecto metálico)

Questão: como pode um utilizador (ou uma aplicação XML) conhecer o contexto para o qual a tag <prego> foi criada?

Resposta: XML Namespaces

Um namespace assegura que um dado conjunto de tags é interpretado no contexto adequado, através da adição de um prefixo a cada tag. O prefixo do namespace identifica um conjunto de tags por forma a que o significado das tags possa ser compreendido de forma correcta.

Componentes do *namespace*

Um *namespace* tem dois componentes principais:

- a declaração do *namespace*, colocada no início do documento XML
- o prefixo do namespace aplicado a cada tag que lhe pertença

Ex: `<note xmlns:n.com='http://www.ciwnote.com/XMLns/1.0'>`
`<n.com:blue> Blue notes means tasks </n.com:blue>`

A declaração do namespace ocorre no elemento root do documento, não na declaração XML. Tem início com “xmlns”, que indica que um namespace está a ser declarado. O prefixo do namespace neste exemplo é n.com (poderia ser nc, NC, note_com, n_com, etc...)

Documentos XML válidos

Para ser útil um documento XML não deve ser apenas bem formado. Necessita também de ser **válido**. Um documento XML válido permite satisfazer pelo menos um de quatro requisitos:

- 1) Cada ficheiro XML deve transportar consigo uma descrição do seu próprio formato.
- 2) Permitir o acordo entre diferentes grupos de utilizadores acerca dos formatos dos documentos XML, nomeadamente para troca de informação.
- 3) As aplicações devem poder utilizar um conjunto de regras que definam a estrutura, sintaxe e vocabulário que os documentos recebidos do exterior devem respeitar.
- 4) Efectuar a validação dos próprios dados do utilizador.

Para validar documentos XML, usou-se inicialmente o conceito Document Type Definition (DTD). O DTD indica um conjunto de regras que especificam o modo como os marcadores estão estruturados. Um DTD, declara os elementos XML válidos, a sua estrutura, e as entidades XML que podem ser usadas por uma classe XML. Limitações e problemas surgidos com o uso dos DTD levaram o W3C a criar a especificação **XML Schema**, que se tornou uma W3C Recommendation em Maio de 2001.

XML Schemas

Os DTD são utilizados para validar documentos XML. No entanto, têm algumas limitações. Duas das mais importantes são:

- 1) Os DTD não são escritos em XML e portanto não respeitam as regras às quais deve obedecer um documento XML bem formado.
- 2) Os DTD não são suficientemente robustos na manipulação de dados XML, o que limita a sua extensibilidade.

Foi, então, criada a especificação XML Schema, que permite estender as capacidades dos DTD, nomeadamente na restrição dos dados. É possível encontrar na documentação, a especificação mencionada como XSDL ou XML Schema Definition Language.

Um XML Schema é um documento XML que define quais são os elementos e as estruturas permitidas nos documentos XML.

XML Schemas

Um XML schema pode ser desenhado de muitas maneiras diferentes.

Nos casos em que um XML schema vai validar informação muito complexa, é possível separá-lo em vários ficheiros para melhorar a sua gestão. Portanto, um XML schema não necessita ser composto por um único ficheiro. Pode ser constituído por um conjunto de documentos XML individuais. Esta possibilidade é particularmente útil nos casos em que a informação a validar pode ser dividida de forma lógica em segmentos ou módulos.

As duas primeiras linhas de um XML schema assemelham-se às do seguinte exemplo:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsd:schema xmlns:xsd="http://www.w3.org.org/2001/XMLSchema">
```

Na segunda linha pode observar-se a tag de abertura do elemento root `<xsd:schema>` contendo a declaração de namespace.

Um documento XML que especifique um schema, ou uma sua parte, tipicamente tem uma extensão `.xsd`

XSL, folhas de estilo

Extensible Stylesheet Language, (XSL) é uma tecnologia da família XML, que tem como objectivo controlar a aparência dos documentos XML. É uma linguagem de folhas de estilos. Estas, especificam regras para a apresentação de uma dada classe de documentos XML.

Uma folha de estilos contém descrições acerca do modo como os elementos XML podem ser formatados por um browser que suporte XML.

Uma folha de estilos é também um documento XML, cuja estrutura é semelhante à do seguinte exemplo:

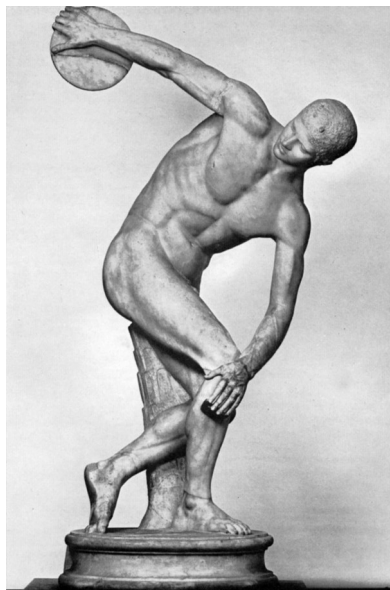
```
<?xml version=" 1.0"?>  
<xsl:stylesheet version="1.0"  
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
```

...instruções xsl aqui...

```
</xsl:stylesheet>
```

Um documento XML que especifique uma folha de estilos, tipicamente, tem uma extensão .xsl

Exercícios



Exercícios XML

- Nos metadados do SNIG, consultar o XML dos registos e:
 - Identificar elementos, atributos e *tags*;
 - Identificar elementos pais/ filhos;
 - Identificar os vários *namespaces* existentes;
 - Descobrir a localização de um XML Schema.
- Ver diferentes visualizações do mesmo ficheiro XML

Informação adicional

Livros:

Ray, Eric T. (2001) “Learning XML”, O’ Reilly, Paris.

Harold, Eliote Rusty e Means, W. Scott (2001) “XML in a Nutshell”, Paris.

Recursos online:

<http://www.w3.org/XML/>

<http://www.w3schools.com/xml/default.asp>

<http://www.w3.org/TR/xml11/>

PPT adaptado de original de António Jorge Silva